仓库监控系统项目总结报告

1. 项目基本功能实现

视频端需具备推拉流传输视屏，检测人脸图像并报警。

后端需具备用户登录注册管理，入侵报警存储。

前端应显示监控图像，可查看入侵记录。

以上基本功能本项目均以实现。

1. 项目拓展功能实现
2. 使用邮件验证码的注册功能
3. 使用Mongodb，入侵自动截图并回放
4. 入侵数据统计分析
5. 人脸比对检测报警，人脸注册，白名单控制
6. 团队合作情况及个人贡献比

（1）18% 付柏逢 20271006 前后端连接，通过sseemitter实现视频，数据库mysql和mongodb的设计与实现，部署云端redis数据库实现邮箱验证码的发送，service层和controller层的实现。

（2）18% 张钰杰 20301061 部分前后端的连接，页面跳转，登录后端实现，用户信息管理、权限修改后端实现，入侵记录管理，查看相对应的入侵图片后端实现，入侵回放功能后端实现，白名单管理、人脸注册后端实现。

（3）16% 张业瑞 20301060 管理员与普通成员界面的设计，普通成员界面的实现，注册界面设计与开发，点击查看单张入侵截图、人脸照片前端实现，点击入侵日期查看入侵截图轮播图前端实现，用户信息及权限编辑、用户删除前端实现，vue组件的搭建与接口设计，系统设计文档撰写。

（4）16% 王俊铭 20301054 管理员界面的设计与实现，登录界面设计与开发

前端分页功能、调用摄像头功能的实现，vue组件的搭建与接口设计，系统实现文档撰写。

（5）16% 周芷平 20301063 人脸识别与人脸检测算法实现，自动报警截图实现，算法优化，组织项目工作以及答辩。

（6）16% 王地 20301053 搭建Centos8+Nginx+Rtmp云端反向代理服务器推拉流实现，协助完成人脸检测&识别算法实现，算法优化，测试&撰写测试文档。

1. 个人工作总结

付柏逢 20271006

1.工作内容

本次的项目中，我主要负责：

1. redis云端数据库的搭建以及实现了邮件验证码功能
2. 前后端的连接，即实现了service和controller层
3. 视频前端的展示，通过springboot的SseEmitter进行快速传照片实现了视频
4. 代码的整合以及一部分的测试
5. 主要问题

对于前期在ecs服务器上搭建redis数据库出现了问题，在redis的配置文件中无法修改密码，最后只能通过命令行进行临时修改且将服务一直保持开启

在后期视频实现的时候，采用将图片编码为base64形式的字符串传到前端进行解码，如何动态修改图片路径以实现图片的快速刷新是个重要难题，最后通过src路径的设置解决

1. 个人心得体会

本次的开放我们选择了java的soringboot框架来实现，虽然前后端的连接较为容易，但前端视频的实现着实不简单，个人觉得写代码应要具有多样性和灵活性，不能因为自己的技术局限就放弃其他的新型技术，在以后的项目中应该多多尝试其他技术，博采众长。

王地 20301053

1.工作内容

在本次项目的开发中，我负责的是视频端的开发工作，涉及到视频流的推流和拉流，云服务部署，以及视频处理，人脸识别算法等部分。

本项目采用的是已经封装好的opencv库，并使用java的API接口去调用，在设计算法中，我了解了三种常用的图像特征提取方法Haar、HOG、LBP，最终我们选择了Haar来作为我们的人脸特征提取算法，采用的是opencv中已经训练好的Haar级联分类器。在这过程中，对Opencv的核心类mat（矩阵）有了更加深入的了解。

在搭建转发服务器的过程中，本项目采用了CentOS8+Nginx+Rtmp的方式，在此过程中熟悉了对Linux的一些操作，又因为传输的是视频流，在翻阅资料的过程中对Rtmp协议， Rtsp协议，以及反向代理服务器有了一定的认识。

在对处理将视频流传输到前端的过程中，最开始尝试了使用FFMmpeg进行H264或X264编码，因此对FFmpeg和这两种编码格式有了一定的学习。但因为FFmpeg是采用C++封装的，而本项目主要是采用Java开发，在寻找使用java调用C++本地库的时候了解了JNI技术以及使用了此项技术的Javacpp库。但最终由于能力以及开发时间的问题没有采用这种方案，最终采用了将视频流逐帧处理编码成Base64的格式发送到前端，然后在前端解码刷新的方案。因而对Base64编码格式有了初步认识。

2.遇到的问题

在开发的过程中，明显感觉到自身的实践能力和工程经验不足，在一定程度上拖慢了项目的开发进度。此外还因为对java的Springboot开发框架不熟悉导致一些开发困难。还有对人脸检测+识别等人工智能相关领域的基础知识掌握不劳导致在开发并优化算法时有些吃力。

3.总结

本次开发经历为我积累了关于图像处理、视频处理、人工智能相关应用的开发经验，提高了我的实践水平，我也因此对计算机视觉研发领域有了一定的兴趣。

张钰杰 20301061

1.主要工作

（1）部分前后端的连接，页面跳转，登录后端实现

（2）用户信息管理、权限修改后端实现

（3）入侵记录管理，查看相对应的入侵图片后端实现

（4）入侵回放功能，加载相应日期入侵图片到日历后端实现

（5）人脸注册、白名单管理后端功能，加载相应用户照片后端实现

2.具体问题

前期实现登录功能，涉及到根据用户权限判断跳转到相应的页面，以及登录验证的工作。后端搜索userid，将相应用户权限等级返回到前端，前端通过判断决定跳转到相应页面。

中期实现用户管理以及用户权限管理、入侵记录管理的功能，这里涉及到数据库的中的增删改查功能，前端axios方法中调用后端controller类中相应的映射方法，controller中调用service类中的方法，service调用mapper中的方法，mapper调用mybatis加载的mapper.xml中的方法，最后利用到数据库中的sql语句实现对user表中数据的增删改查。同理，入侵记录管理后端实现，通过操作userfile表，实现表中数据的操作，记录管理增加了查看相应图片的功能，将表中的记录名返回到前端，之后在后端通过记录名获取相应图片，并获取完整图片名，之后根据文件路径进行base64编码，将编码后的src返回到前端，前端通过src显示相应图片。开发中期进行了搭建mongoDB数据库，同时配合mysql数据库进行管理图片。

后期实现入侵回放、人脸注册和白名单功能。入侵回放的前端用到了日历，每一天都有相应的轮播图实现图片集的展示，于是我在前端编写axios方法获取相应的日期，传到后端，后端通过日期查找对应的图片集合，每查找到一张，便进行进行base64编码，并存到结果集数组中，最后将结果集数组传到前端。前端便通过src数组，显示对应日期的入侵图片集。人脸注册功能，通过调用摄像头拍摄人脸照片，在获取用户输入的userid后，保存图片至本地，图片名通过用户名来命名，用于标识。白名单管理中涉及到用户信息的加载，这里和用户管理功能类似，不同的是，这里通过获取对应行的userid来查看相应的用户照片，另外在删除功能中，不仅涉及到数据库中该用户记录的删除，还在后端实现相应用户照片的删除。

3.总结

工作过程中遇到一些困难，开发前期对springboot框架不够熟悉，以及前端axios中的get和post映射方法不熟，中期在数据库中数据加载到对应表格的实现中遇到了较大困难，后期由于热部署问题，图片无法实时更新。这些问题后来都得到了解决，这次的项目让我更加熟悉了springboot后端框架以及前后端的连接，熟练了对数据库的操作，学到了图片上传、下载、显示等的技能。同时也增强了自信，提高了团队意识。

王俊铭 20301054

1.工作内容

在本次项目的开发中，我负责的部分是前端页面的设计。由于此前的学习从未学习过有关前端设计的知识所以我开展工作的第一步是先学习前端的相关知识。

我先花了两天半的时间通过教学视频和菜鸟教程学习了html+css+javascript+vue的基本知识。后面配合搭档一起实现了第一版的页面，虽然十分粗糙但是有了大致的全部框架。

后续发现如果仅仅的依靠html+css+javascript+vue来设计实现页面将会使得前端代码量过大且某些功能实现起来过于困难，因此我们需要寻找更加便捷的方法来设计前端页面。

在搭档的帮助下我们找到了element UI组件库，element UI基于vue设计并且封装了许多如日历、下拉菜单这样的组件，极大的节约了开发工时不需要再费尽心思的去设计css样式，在element UI的加持下我们只需要设计页面布局和vue实例的构建由此得到了第二版也就是最终版页面。

我们差不多在11号就结束了前端页面的设计与实现，此后的几天为前后端的对接提供建议。

2.遇到的问题

在开发的过程中，由于接触的是一门完全陌生的开发语言所以需要大量的时间去学习，但速成的成果并不能完全的满足开发需求因此中途出现了开发困难的问题。

3.总结

本次开发经历为我积累了关于页面设计的开发经验，提高了我的实践水平，我也因此对前端开发领域有了浓厚的兴趣。

张业瑞 20301060

1.工作内容

我使用html+css进行了注册界面的开发，使用网页原型设计工具墨刀进行了管理员界面以及普通成员界面的设计，使用vue框架与element UI组件库进行了管理员以及普通成员界面的开发。在界面开发完毕以后，我和我的搭档开始给网页中的按钮添加事件、添加编辑方法，以及给表格界面添加了分页功能，点击查看单张入侵截图、人脸照片的前端实现，点击入侵日期查看入侵截图轮播图的前端实现，以及用户信息及权限编辑、用户删除的前端实现

2.遇到的问题

我和我的搭档都对前端编程知识不太了解，所以我们只能从0开始学习前端编程知识，我们用了两天去学习html与css的基础知识，好在网络上有很多比较好的教程，所以我们也比较轻松地学习了html与css的基础知识。Html+css开发注册登录页面还比较简单，而仅仅使用html与css开发管理员与普通成员界面对于我们来说则有些困难了，好在我们学会使用了element UI的组件库，成员界面的开发也相对简单一些了。

3.总结

通过本次项目开发，我对前端编程语言更加熟悉，也对前端开发更加有兴趣。这次项目开发过程也让我收获很多，我也体会到了团队开发的过程，很感谢我的队友们的帮助。虽然开发过程并不一直很顺利，但我们最后还是成功完成了项目开发，这两个星期对我来说是一段不错的经历。

周芷平 20301063

1.主要工作

作为组长，我的主要内容是统筹小组的工作，进行模块工作的划分，让项目进度匀速前进。我的个人工作内容在与视频端的开发。我负责视屏获取，人脸识别的算法，与人脸图像的比对。

梳理整个项目流程。我的算法端应该做到实时监控视屏内容，将人脸框选，并且将框选的图片与本地的白名单中的人脸图片进行对比，输出一个相关系数值。如果此值低于我们设定的值则报警处理，并自动截取画面将图片存入我们的库中。具体实现是依靠opencv中自带的模型文件，将人脸框选出来，进行灰度处理，输出成灰度直方图，同样将已有的白名单中的人脸图片输出成灰度直方图进行比对。将相关系数低的帧进行警告并截图

2.主要问题

在项目中，我们遇到的问题主要有算法处理的速度太慢，算法精度不够好。在处理速度方面通过优化循环，处理算法内容将速度提升，当然处理速度也与硬件有很多关系，提升硬件也是一种方式。算法精度问题较难解决，但总体能让摄像头中的人脸与他的照片达到0.6往上的相关系数。如果要优化算法，需要改变我们的识别流程与结构，或许要重新训练模型，但考虑我们的项目工期只好放弃了。

在视屏传输前端的过程中，同样也遇到了很大的问题，我们一开始希望我们使用视频流将处理好的图片写成视屏流传输，但遇到了技术难题，很难将内存中的连续图片处理成视频流。于是采取逐帧传输，由于不熟悉springboot的操作方式，只能协助后端人员进行编写，提供接口和思路。

3.心得

作为组长组织了本次的项目开发，进行了视频端的研发工作。本次项目中同学们都很给力的配合，基本在deadline前都做好了开发，我们的项目开发速度较快也比较平稳。在本次项目中增加了很多作为项目经理的经验，也对opencv进行了学习，增长了知识，收货颇丰。